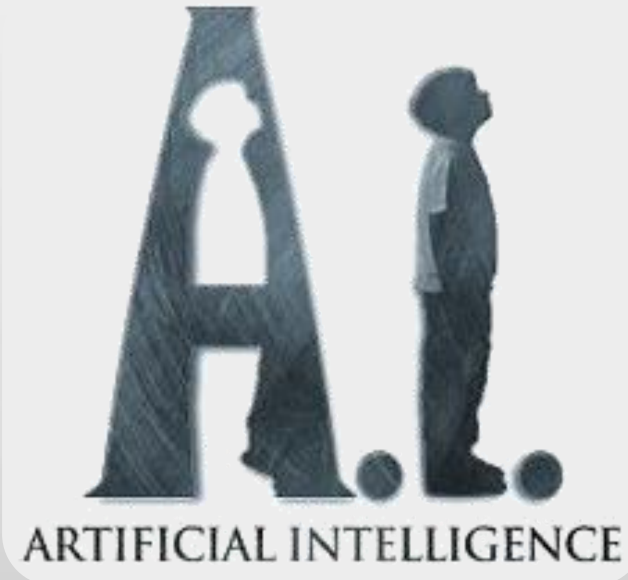
First-Order Logic

Chapter 8



NORTH-WEST UNIVERSITY
YUNIBESITI YA BOKONE-BOPHIRIMA
NOORDWES-UNIVERSITEIT
INSTITUTIONAL OFFICE

Overview of lecture

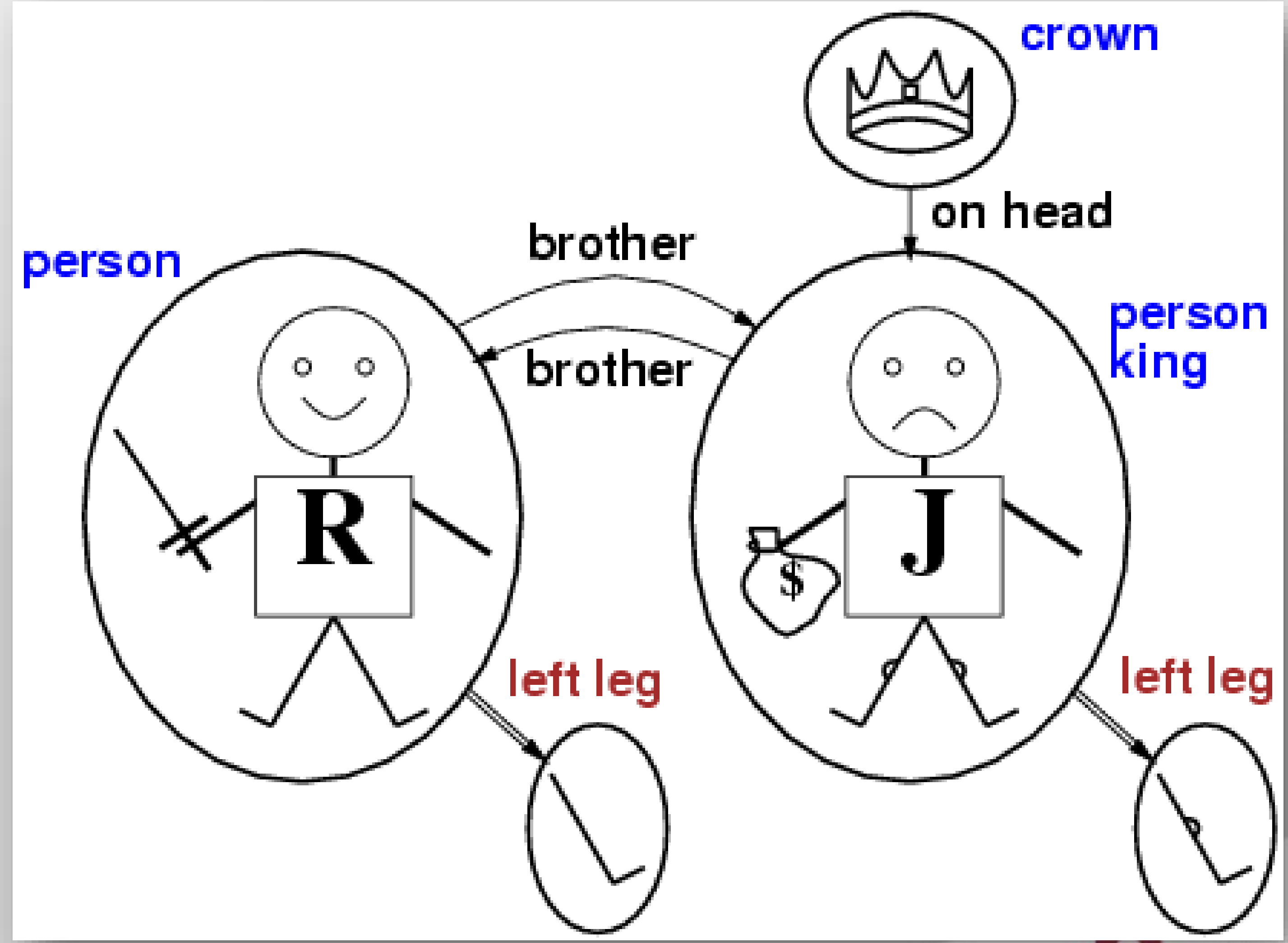
- Quantifiers
- Equality
- Using First-order logic

Universal quantification (∀)

1189 to 1199

1199 to 1215

Universal quantification (∀)



Quantifiers

- Let us express properties of sets of objects
- First-order logic contains two standard quantifiers
  - Universal quantifier ($\forall$)
  - Existential quantifier ($\exists$)

Universal quantification (∀)

- Difficult to express general rules in proposition logic
- "All kings are persons" in first-order logic

  $\forall x\ King(x) \Rightarrow Person(x)$
- $\forall x\ P$
- True in all possible extended interpretations

Universal quantification (∀)

- $\forall x\ King(x) \Rightarrow Person(x)$

  $x \rightarrow$ Richard the Lionheart
  $x \rightarrow$ King John
  $x \rightarrow$ Richard's left leg
  $x \rightarrow$ John's left leg
  $x \rightarrow$ the crown

Universal quantification (∀)

- $\forall x\ King(x) \Rightarrow Person(x)$

  Richard the Lionheart is a king ⇒ Richard the Lionheart is a person

  King John is a king ⇒ King John is a person

  Richard’s left leg is a king ⇒ Richard’s left leg is a person

  John’s left leg is a king ⇒ John’s left leg is a person

  The crown in a king ⇒ the crown is a person

Universal quantification (∀)

- Implication (Figure 7.8)

P	Q	$P \Rightarrow Q$
false	false	true
false	true	true
true	false	false
true	true	true

Existential quantification (∃)

- Universal quantifier makes statements about each object
- Existential quantification make statements about an object or some objects
- For example

  $\exists x\ Crown(x) \wedge OnHead(x, John)$
- $\exists x\ P$
- True in at least one extended interpretation

Existential quantification (∃)

- $\exists x$ Crown(x) ∧ OnHead(x, John)

Richard the Lionheart is a crown ∧ Richard the Lionheart is on John's head

King John is a crown ∧ King John is on John's head

Richard's left leg is a crown ∧ Richard's left leg is on John's head

John's left leg is a crown ∧ John's left leg is on John's head

The crown is a crown ∧ the crown is on John's head

Nested Quantifiers

- $\forall x\ \forall y\ Brother(x,y) \Rightarrow Sibling(x,y)$
- $\forall x,y\ Sibling(x,y) \Leftrightarrow Sibling(y,x)$
- Everybody loves somebody

  $\forall x\ \exists y\ Loves(x,y)$
- There is someone who is loved by everyone

  $\exists y\ \forall x\ Loves(x,y)$
- Confusion can arise

  $\forall x\ [Crown(x) \vee (\exists x\ Brother(Richard, x))]$

Connections between ∀ and ∃

- The two quantifiers connected to each other through negation

$\forall x\ \neg Likes(x, Parsnips)$ equivalent to
$\neg\exists x\ Likes(x, Parsnips)$

Connections between ∀ and ∃

$\forall x$ Likes(x, IceCream) equivalent to

$\neg\exists x\ \neg$Likes(x, IceCream)

Connections between ∀ and ∃

- De Morgan's rules for quantified sentences:

$$\forall x\ \neg P \equiv \neg \exists x\ P$$

$$\neg \forall x\ P \equiv \exists x\ \neg P$$

$$\forall x\ P \equiv \neg \exists x\ \neg P$$

$$\exists x\ P \equiv \neg \forall x\ \neg P$$

Equality

- Father(John) = Henry
- $\exists$ x,y Brother(x, Richard) $\wedge$ Brother(y, Richard) $\wedge$ $\neg$(x=y)
- $\neg$(x=y) written as (x$\neq$y)

- This is not the same as:

  $\exists$ x,y Brother(x, Richard) $\wedge$ Brother(y, Richard)

Using first-order logic

- Consider examples of how to use first-order logic
- Represent simple domains
- Sentences are added to the knowledge base with TELL and called assertions
- TELL(KB, King(John))

  TELL(KB, $\forall x$ King(x) $\Rightarrow$ Person(x))

Using first-order logic

- Ask questions to knowledgebase with ASK

  ASK(KB, King(John))

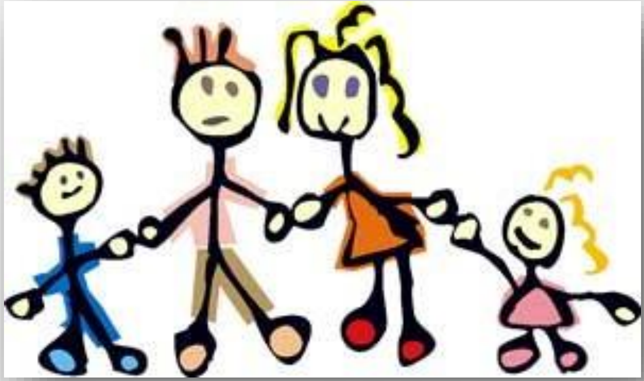
  ASK(KB, Person(John))

  ASK(KB, $\exists x$ Person(x))

  ASKVARS(KB, Person(x))
- Answer is substitution or binding list {variable/term}, for example {x/John}

Using first-order logic

- The kinship domain
  - Objects in domain are people
  - Properties: Male and Female
  - Binary relations: Parent, Sibling, Brother, Sister, Child, Daughter, Son, Spouse, Wife, Husband, Grandparent, Grandchild, Cousin, Aunt, Uncle
  - Functions: Mother en Father



Using first-order logic

- The kinship domain

  $\forall m,c\ Mother(c) = m \Leftrightarrow Female(m) \land Parent(m,c)$

  $\forall w,h\ Husband(h, w) \Leftrightarrow Male(h) \land Spouse(h, w)$

  $\forall x\ Male(x) \Leftrightarrow \neg Female(x)$

  $\forall p,c\ Parent(p,c) \Leftrightarrow Child(c,p)$

  $\forall g,c\ Grandparent(g,c) \Leftrightarrow \exists p\ Parent(g,p) \land Parent(p,c)$

  $\forall x,y\ Sibling(x,y) \Leftrightarrow x \neq y \land \exists p\ Parent(p,x) \land Parent(p,y)$

Using first-order logic

- The kinship domain
  - **Axioms** are basic factual information from which useful conclusions can be derived
  - **Definitions** have the form $\forall x,y\ P(x,y) \Leftrightarrow \ldots$
  - **Theorems** are entailed by the axioms, e.g. $\forall x,y\ Sibling(x,y) \Leftrightarrow Sibling(y,x)$
  - From logical point of view: only axioms in knowledgebase
  - From practical point of view: axioms and theorems in knowledgebase

Using first-order logic

- The kinship domain
- Not all axioms are definitions

  $\forall x\ Person(x) \Leftrightarrow \ldots$

  $\forall x\ Person(x) \Rightarrow \ldots$

  $\forall x\ \ldots \Rightarrow Person(x)$

  – Axioms can also be facts

  Male(Jim) and Spouse(Jim, Laura)

Using first-order logic

- Basic set of axioms

$\forall m,c\ Mother(c) = m \Leftrightarrow Female(m) \wedge Parent(m,c)$

$\forall w,h\ Husband(h, w) \Leftrightarrow Male(h) \wedge Spouse(h, w)$

$\forall x\ Male(x) \Leftrightarrow \neg Female(x)$

$\forall p,c\ Parent(p,c) \Leftrightarrow Child(c,p)$

$\forall g,c\ Grandparent(g,c) \Leftrightarrow \exists p\ Parent(g,p) \wedge Parent(p,c)$

$\forall x,y\ Sibling(x,y) \Leftrightarrow x \neq y \wedge \exists p\ Parent(p,x) \wedge Parent(p,y)$

Assignment

- Study today’s work
  - Sections 8.2.6 to 8.3.2
  - Also study 8.2.8 (Database semantics)
- Please study for Theory quiz 6 on today’s work
  - Thursday, 21 October 2021
- Protégé…